Convolutional codes from units in matrix and group rings

Ted Hurley

Abstract

A general method for constructing convolutional codes from units in Laurent series over matrix rings is presented. Using group rings as matrix rings, this forms a basis for in-depth exploration of convolutional codes from group ring encodings, wherein the ring in the group ring is itself a group ring. The method is used to algebraically construct series of convolutional codes. Algebraic methods are used to compute free distances and to construct convolutional codes to prescribed distances.

1 Introduction

Methods are presented for constructing convolutional codes using units in Laurent series of finite support over matrix rings. By considering group rings as matrix rings, convolutional codes are constructed from units in Laurent series over group rings; these may be considered as group rings over group rings. Thus convolutional codes are constructed by considering a group ring RG where the ring R is itself a group ring.

The methods are based on the general method in [3] for constructing *unit-derived* codes from group rings where now the ring of the group ring is a group ring and the group of the group ring may be an infinite group such as the infinite cyclic group.

For general information on group rings and related algebra see [9].

Using these algebraic methods, the range of convolutional codes available is expanded and series of convolutional codes are derived. Free distances and codes to a prescribed free distances may also be derived. Indeed many of the existing convolutional codes can be obtained in the manner of this paper.

The paper [8] is an often quoted source of information on convolutional codes wherein is mentioned the lack of algebraic methods for constructing convolutional codes; and that many of the existing ones have been found by computer search and are of necessity of relatively short memories.

The methods are fairly general and use properties of group rings and their embedding into matrix rings. Zero-divisors and units in group rings enables the construction of units in certain polynomial rings and/or group rings over these group rings from which convolutional codes can be constructed. Properties of the convolutional codes can be studied and derived from properties of group rings. In many instances the free distances can be calculated algebraically and convolutional codes to a specified free distance, as for example in Theorem 7.3 or Theorem 14.1 below, can be constructed.

The following are some of the applications of the general method and these in themselves constitute new methods for constructing convolutional codes:

- The construction of series of binary $(2,1)$ convolutional codes and calculation of their free distances using the group ring $(FC_2)C_\infty$ where F is a field of characteristic 2;
- Given a linear cyclic code $\mathcal{C}$ with $d = \min(d_1, d_2)$ where d_1 is the minimum distance of $\mathcal{C}$ and d_2 is the distance of the dual of $\mathcal{C}$, the generator polynomial f of $\mathcal{C}$ is mimicked in RC_∞ to construct convolutional $(2,1)$ codes of minimum free distance $d+2$;
- The construction of rate $\frac{3}{4}$ and higher rate convolutional codes with prescribed minimum distance;
- The construction of convolutional codes over a field F of characteristic p for any prime p using nilpotent elements in the field FG where G is a group whose order is divisible by p;
- The construction of *Hamming type* convolutional codes and calculating their free distances; the construction of Hamming-type convolutional codes to a desired minimum free distance;

- The construction of convolutional codes using idempotents in group rings. These are particularly used in cases where the characteristic of the field does not divide the order of the group; *characters* of groups and *character tables* come into play in constructing these convolutional codes.

1.1 Algebraic Description of Convolutional Codes

Background on general algebra and group rings may be obtained in [9].

For any ring R, $R[z]$ denotes the polynomial ring with coefficients from R and $R_{r\times n}$ denotes the ring of $r \times n$ matrices with coefficients from R. R^n is used to denote $R_{1\times n}$ and thus $R^n = \{(r_1, r_2, \ldots, r_n) : r_i \in R\}$.

It is easy to verify that $R_{r\times n}[z] \cong R[z]_{r\times n}$.

$R[z, z^{-1}]$ is used to denote the set of Laurent series of finite support in z with coefficients from R. *Finite support* means that only a finite number of the coefficients are non-zero. It is clear that $R[z, z^{-1}] \cong RC_\infty$, where C_∞ denotes the infinite cyclic group. (Elements in group rings have finite support.)

Note also the relationship between $R[z]$ and RC_∞ – $R[x] \cong T$ where T denotes the algebra of *non-negative elements*, i.e. the algebra of elements $w = \sum_{i=0}^{\infty} \alpha_i g^i$, in RC_∞.

If $\mathbb{F}$ is an integral domain then $\mathbb{F}[z]$ has no zero-divisors and only trivial units – the units of $\mathbb{F}[z]$ are the units of $\mathbb{F}$.

See [8] and/or [1] for basic information on convolutional codes and algebraic descriptions are described therein. The (equivalent) algebraic description given in [2] is extremely useful and is given below.

A convolutional code $\mathcal{C}$ of length n and dimension k is a direct summand of $\mathbb{F}[z]^n$ of rank k. Here $\mathbb{F}[z]$ is the polynomial ring over $\mathbb{F}$ and $\mathbb{F}[z]^n = \{(v_1, v_2, \ldots, v_n) : v_i \in \mathbb{F}[z]\}$.

Suppose V is a submodule of $\mathbb{F}[z]^n$ and that $\{v_1, \ldots, v_r\} \subset \mathbb{F}[z]^n$ forms a generating set for V. Then $V = \text{Image}\, M = \{uM : u \in \mathbb{F}[z]^r\}$ where $M = \begin{bmatrix} v_1 \\ \vdots \\ v_r \end{bmatrix} \in \mathbb{F}[z]_{r\times n}$. This M is called a *generating matrix* of V.

A generating matrix $G \in \mathbb{F}[z]_{r\times n}$ having rank r is called a *generator* or *encoder matrix* of $\mathcal{C}$.

A matrix $H \in \mathbb{F}[z]_{n\times(n-k)}$ satisfying $\mathcal{C} = \ker H = \{v \in \mathbb{F}[z]^n : vH = 0\}$ is said to be a *control matrix* of the code $\mathcal{C}$.

2 Convolutional codes from units

Let R be a ring which is a subring of the ring of matrices $F_{n\times n}$.

In particular the group ring FG is a subring of $F_{n\times n}$, where $n = |G|$, by an explicit embedding given in [4]. There is no restriction on F in general but it is assumed to be a field here; however many of the results will hold more generally.

Units and *zero-divisors* in any ring are defined in the usual manner.

Construct R-convolutional codes as follows:

2.1 Polynomial case

For clarity the polynomial case is considered initially although this is a special case of the more general construction.

Suppose $f(z)g(z) = 1$ in $R[z]$. Essentially then the encoder matrix is obtained from $f(z)$ and the decoder or control matrix is obtained from $g(z)$ using a variation on the method for constructing unit-derived codes as formulated in [3] for non-singular matrices.

Now $f(z) = (f_{i,j}(z))$ is an $n \times n$ matrix with entries $f_{i,j}(z) \in F[z]$. Similarly $g(z) = (g_{i,j}(z))$ is an $n \times n$ matrix over $F[z]$. Suppose $r[z] \in F[z]^r$ and consider $r[z]$ as an element of $\mathbb{F}[z]^n$ (by adding zeros

to the end of it). Then define a mapping $\gamma : F[z]^r \to F[z]^n$ by $\gamma : r(z) \mapsto r(z)f(z)$. The code $\mathcal{C}$ is the image of γ. Since $r[z]$ has zeros in its last $(n-r)$ entries as a member of $F[z]^n$, this means that *the generator matrix is the first* r *rows* of $f(z)$ which is an $r \times n$ matrix over $F[z]$. Since $f(z)$ is invertible, this generator matrix has rank r and is thus the encoder matrix which we denote by $G(z)$. For this polynomial case, $G(z)$ is a basic generator matrix – see A.1 Theorem in [8].

$w(z) \in \mathbb{F}[z]^n$ is a codeword if and only if $w(z)g(z)$ is in $\mathbb{F}[z]^r$, that is, if and only if the final $(n-r)$ entries of $w(z)g(z)$ are all 0. Suppose $w(z) = (\alpha_1(z), \alpha_2(z), \ldots, \alpha_n(z))$. Then this condition is that

$$(\alpha_1(z), \alpha_2(z), \ldots, \alpha_n(z)) * \begin{pmatrix} g_{1,r+1}(z) & g_{1,r+2}(z) & \ldots & g_{1,n}(z) \\ g_{2,r+1}(z) & g_{2,r+2}(z) & \ldots & g_{2,n}(z) \\ \vdots & \vdots & \vdots & \vdots \\ g_{n,r+1}(z) & g_{n,r+2}(z) & \ldots & g_{n,n}(z) \end{pmatrix} = 0$$

The check or *control matrix* $H(z)$ of the code is thus:

$$\begin{pmatrix} g_{1,r+1}(z) & g_{1,r+2}(z) & \ldots & g_{1,n}(z) \\ g_{2,r+1}(z) & g_{2,r+2}(z) & \ldots & g_{2,n}(z) \\ \vdots & \vdots & \vdots & \vdots \\ g_{n,r+1}(z) & g_{n,r+2}(z) & \ldots & g_{n,n}(z) \end{pmatrix}$$

This has size $n \times (n-r)$ and is the matrix consisting of the last $(n-r)$ columns of $g(z)$ or in other words the matrix obtained by deleting the first r columns of $g(z)$.

Since $f(z), g(z)$ are units, it is automatic that $\operatorname{rank} G(z) = r$ and $\operatorname{rank} H(z) = (n-r)$.

2.1.1 Restatement of polynomial case

Suppose then $f(z)g(z) = 1$ in $R[z]$. The set-up may be restated as follows:

$$f(z) = \begin{pmatrix} f_1(z) \\ f_2(z) \end{pmatrix}$$

$$g(z) = \begin{pmatrix} g_1(z), & g_2(z) \end{pmatrix}$$

where $f_1(z)$ is an $r \times n$ matrix, $f_2(z)$ is an $(n-r) \times n$ matrix, $g_1(z)$ is an $n \times r$ matrix and $g_2(z)$ is an $n \times (n-r)$ matrix.

Then $f(z)g(z) = 1$ implies

$$\begin{pmatrix} f_1(z) \\ f_2(z) \end{pmatrix} \times \begin{pmatrix} g_1(z), & g_2(z) \end{pmatrix} = 1$$

Thus

$$\begin{pmatrix} f_1(z)g_1(z) & f_1(z)g_2(z) \\ f_2(z)g_1(z) & f_2(z)g_2(z) \end{pmatrix} = 1$$

From this it follows that
$f_1(z)g_1(z) = I_{r \times r}$,
$f_1(z)g_2(z) = 0_{r \times (n-r)}$,
$f_2(z)g_1(z) = 0_{(n-r) \times r}$,
$f_2(z)g_2(z) = I_{(n-r) \times (n-r)}$.

Thus $f_1(z)$ is taken as the generator or encoder matrix and $g_2(z)$ is then the check or control matrix. Note that both $f_1(z), f_2(z)$ have right finite support inverses and thus by Theorem 6.3 of [8] the generator matrix f_1 is noncatastrophic.

Given $f(z)g(z) = 1$ by the general described method of unit-derived code in [3] a convolutional code can be constructed using *any* rows of $f(z)$. If rows $\{j_1, j_2, \ldots, j_r\}$ are chosen from $f(z)$ then we get an encoding $F^r[z] \to F^n[z]$ with generator matrix consisting of these r rows of $f(z)$ and check/control matrix is obtained by deleting the $\{j_1, j_2, \ldots, j_r\}$ columns of $g(z)$.

Cases with $f(z)g(z) = 1,\ f(z), g(z) \in R[z, z^{-1}]$, will also in a similar manner produce convolutional codes. The next section, Section 2.2, describes the similar process for these in detail.

2.2 More generally

Let $f(z, z^{-1}), g(z, z^{-1}) \in R[z, z^{-1}]$ be such that $f(z, z^{-1})g(z, z^{-1}) = 1$.

Suppose now

$$f(z, z^{-1}) = \begin{pmatrix} f_1(z, z^{-1}) \\ f_2(z, z^{-1}) \end{pmatrix}$$

$$g(z, z^{-1}) = \begin{pmatrix} g_1(z, z^{-1}), & g_2(z, z^{-1}) \end{pmatrix}$$

where $f_1(z, z^{-1})$ is an $r \times n$ matrix, $f_2(z, z^{-1})$ is an $(n-r) \times n$ matrix, $g_1(z, z^{-1})$ is an $n \times r$ matrix and $g_2(z, z^{-1})$ is an $n \times (n-r)$ matrix.

Then

$$\begin{pmatrix} f_1(z, z^{-1}) \\ f_2(z, z^{-1}) \end{pmatrix} \times \begin{pmatrix} g_1(z, z^{-1}), & g_2(z, z^{-1}) \end{pmatrix} = 1$$

Thus

$$\begin{pmatrix} f_1 g_1 & f_1 g_2 \\ f_2 g_1 & f_2 g_2 \end{pmatrix} = 1$$

From this it follows that
$f_1(z, z^{-1})g_1(z, z^{-1}) = I_{r\times r}$,
$f_1(z, z^{-1})g_2(z, z^{-1}) = 0_{r\times(n-r)}$,
$f_2(z, z^{-1})g_1(z, z^{-1}) = 0_{(n-r)\times r}$,
$f_2(z, z^{-1})g_2(z, z^{-1}) = I_{(n-r)\times(n-r)}$.

Thus $f_1(z, z^{-1})$ is taken as the generator or encoder matrix and $g_2(z, z^{-1})$ is then the check or control matrix. It is seen in particular that $f_1(z, z^{-1}), f_2(z, z^{-1})$ have right finite support inverses and thus by Theorem 6.6 of [8] the generator matrix f_1 is noncatastrophic.

Given $f(z, z^{-1})g(z, z^{-1}) = 1$ by the general described method of unit-derived code of [3] codes *any* rows of $f(z, z^{-1})$ can be used to construct a convolutional. If rows $\{j_1, j_2, \dots, j_r\}$ are chosen from $f(z, z^{-1})$ then an encoding $F^r[z] \to F^n[z]$ is obtained with generator matrix consisting of these r rows of $f(z)$ and check/control matrix obtained by deleting the $\{j_1, j_2, \dots, j_r\}$ columns of $g(z)$.

2.2.1 Particular case

Suppose $f(z)g(z) = z^t$ in $R[z]$. Then $f(z)(g(z)/z^t) = 1$. Now $(g(z)/z^t)$ involves negative powers of z but has finite support. The encoder matrix is obtained from $f(z)$ and the decoder or control matrix is obtained from $(g(z)/z^t)$ using the method as formulated in Section 2.2. It is also possible to consider $(f(z)/z^i)(g(z)/z^j) = 1$ with $i + j = t$ and to derive the generator matrix from $(f(z)/z^i)$ and the check/control matrix from $(g(z)/z^j)$.

The control matrix contains negative powers of z but a polynomial control matrix is easy to obtain from this.

Note that z^{-n} are units worth considering in $R[z, z^{-1}]$ but that other elements in $R[z, z^{-1}]$ may have inverses with infinite support and the inverses are thus outside $R[z, z^{-1}]$. However in some cases $\sum_{i=-t}^{m} \alpha_i z^i \in R[z, z^{-1}]$ has an inverse in $R[z, z^{-1}]$, for example in certain cases when the α_i are nilpotent, and here also convolutional codes may be defined with (direct) noncatastrophic generator matrices. All these are cases of $f(z, z^{-1}) \times g(z, z^{-1}] = 1 \in R[z, z^{-1}]$ but may be worth considering originally from polynomials for the construction.

2.2.2 Uninteresting zero-divisors

In [3] units and zero-divisors in group rings are used to construct codes. Zero-divisors in $R[z]$ are not too interesting: Suppose $uw = 0$ in $R[z]$ and u is an element of least degree so that $uw = 0$. Then w or

u has degree zero; if w has degree 0 then it is a zero-divisor of each coefficient of u and if u has degree zero then it is a zero-divisor of each coefficient of w.

Thus if we require zero-divisor codes in $R[z]$ we are looking at direct sums of zero-divisor codes in R. Using units in $R[z]$ to construct codes is far more productive.

2.3 Group ring matrices

In the constructions of Section 2.1 or in the more general Section 2.2, R is a subring of $F_{n\times n}$. Suppose now $R = FG$ is the group ring of the group G over F.

The group ring RG is a subring of $F_{n\times n}$ using an explicit correspondence between the group ring RG and the ring of RG-matrices, see e. g. [4].

Thus the methods of Section 2.1 and/or Section 2.2 may be used to define convolutional codes using group rings $R = FG$ as a subring of $F_{n\times n}$ and then forming $R[z, z^{-1}] \cong RC_\infty$, which is the group ring over C_∞ with coefficients from the group ring $R = FG$.

To obtain units in $R[z, z^{-1}]$ (which includes $R[z]$) we are lead to consider zero-divisors and units in $R = FG$.

$R = FG$ is a rich source of zero-divisors, and units, and consequently $R[z, z^{-1}]$ is a rich source of units. There are methods available for constructing units and zero-divisors in FG. If F is a field, every non-zero element of FG is either a unit or a zero-divisor. What is required are units in $R[z]$, where $R = FG$, a group ring, and these can be obtained by the use of zero-divisors and units in R as coefficients of the powers of z.

In what follows bear in mind that in $R[z, z^{-1}]$ it is possible and desirable that R has zero-divisors and units, as when R is a group ring.

3 Convolution codes from group rings

Suppose then $\sum_{i=-m}^{n} \alpha_i z^i \times \sum_{j=-m}^{n} \beta_j z^j = 1$ in the group ring $RC_\infty = R[z, z^{-1}]$ with $\alpha_i \in R$ and C_∞ generated by z. By multiplying through by a power of z this is then $\sum_{i=0}^{n} \alpha_i z^i \times \sum_{j=-m}^{n} \beta_j z^j = 1$.

The case with $m = 0$ gives polynomials over z. Here we have $\sum_{i=0}^{n} \alpha_i z^i \times \sum_{i=0}^{t} \beta_j z^j = 1$ where $\alpha_n \neq 0, \beta_t \neq 0$ and looking at the coefficient of z^0 it is clear that we must also have $\alpha_0 \neq 0, \beta_0 \neq 0$. This can be considered an an equation in RC_∞ with non-negative powers. Solutions may be used to construct convolutional codes.

By looking at the highest and lowest coefficients we then have that $\alpha_0 \times \beta_0 = 1$ and $\alpha_n \times \beta_t = 0$. Thus in particular α_0 is a unit with inverse β_0 and α_n, β_t are zero divisors.

Solutions of the general equation $\sum_{i=0}^{n} \alpha_i z^i \times \sum_{j=-m}^{n} \beta_j z^j = 1$ can also be used to form convolutional codes and polynomial generator matrices may be derived from these.

4 Examples

4.1 A prototype example

Let $R = \mathbb{Z}_2 C_4$. Then $\alpha_0 = a + a^2 + a^3$ satisfies $\alpha_0^2 = 1$ and $\alpha_2 = a + a^3$ satisfies $\alpha_2^2 = 0$.

Thus $w = \alpha_0 + \alpha_1 z + \alpha_2 z^2$ in RC_∞ satisfies $w^2 = \alpha_0\alpha_0 + z(\alpha_0\alpha_1 + \alpha_1\alpha_0) + z^2(\alpha_0\alpha_2 + \alpha_1^2 + \alpha_2\alpha_0) + z^3(\alpha_1\alpha_2 + \alpha_2\alpha_1) + z^4(\alpha_2\alpha_2) = 1 + z^2\alpha_1^2$, since the α_i commute. Now require that $\alpha_1^2 = 0$ and then $w^2 = 1$.

In particular letting $\alpha_1 = \alpha_2$ implies that $w^2 = 1$. However, just to be different, consider $\alpha_1 = 1 + a^2$ and then also $\alpha_1^2 = 0$.

Now α_0 corresponds to the matrix $\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$, α_2 corresponds to the matrix $\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$ and α_1 corresponds to the matrix $\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$.

Take the first two rows of w to generate a convolutional code and then the last two columns of w is the control matrix of this code.

This gives the following generator matrix:

$$G = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} z + \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z^2$$

The control matrix is:

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} z + \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z^2$$

The code has length 4 and dimension 2. It may be shown that the free distance of this code is 6.

This can be generalised.

5 Convolutional codes from nilpotent elements

The following two theorems are useful in constructing new classes of convolutional codes.

Theorem 5.1 *Let $R = FG$ be the group ring of a group G over a field F with characteristic 2. Suppose $\alpha_i \in R$ commute. Let $w = \sum_{i=0}^{n} \alpha_i z^i \in RC_\infty$. Then $w^2 = 1$ if and only if $\alpha_0^2 = 1, \alpha_i^2 = 0, i > 0$.*

Proof: The proof of this is straight-forward and is omitted.

□

The following is a generalisation of Theorem 5.1; its proof is also straight-forward and is omitted.

Theorem 5.2 *Let $R = FG$ be the group ring of a group G over a field F with characteristic 2. Suppose $\alpha_i \in R$ commute. Let $w = \sum_{i=0}^{n} \alpha_i z^i \in RC_\infty$. Then $w^2 = z^{2t}$ if and only if $\alpha_i^2 = 0, i \neq t$ and $\alpha_t^2 = 1$.*

To then construct convolutional codes proceed as follows. Find elements α_i with $\alpha_i^2 = 0$ and units u with $u^2 = 1$ in the group ring R. Then form units in $R[z]$ or $R[z, z^{-1}]$ using Theorem 5.1 or Theorem 5.2. From these units, convolutional codes are defined using the methods described in Section 2.1 or Section 2.2.

5.1 Examples 1

Consider now $\alpha_0 = a + a^2 + a^3$ and for $i > 0$ define $\alpha_i = a + a^3$ or $\alpha_i = 0$ in the group ring $R = \mathbb{Z}_2 C_4$. Then $\alpha_0^2 = 1$ and $\alpha_i^2 = 0, i > 0$. We could also take $\alpha_i = 1 + a^2$.

Define $w(z) = \sum_{i=0}^{n} \alpha_i z^i$ in RC_∞. By Theorem 5.1, $w^2 = 1$.

The matrix corresponding to α_0 is $\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$ and the matrix corresponding to $\alpha_i, i \neq 0$ is $\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$ or else is the zero matrix.

Now specify that the first two rows of w give the generator matrix and from this it follows that the last two columns of w is a control matrix.

This gives the following generator matrix:

$$G = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix} + \delta_1 \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z + \delta_2 \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z^2 + \ldots + \delta_n \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z^n$$

where $\delta_i = 1$ when $\alpha_i \neq 0$ and $\delta_i = 0$ when $\alpha_i = 0$.

The control matrix is:

$$H = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} + \delta_1 \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z + \delta_2 \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z^2 + \ldots + \delta_n \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z^n.$$

The code has length 4 and dimension 2. The free distance is at least 6 for any $n \geq 2$ and in many cases it will be larger. Polynomials used for generating cyclic linear codes suitably converted to polynomials in $R[z]$ prove particularly useful and amenable – see for example Section 7 below.

5.1.1 Particular Example

The $(4, 2)$ convolutional code with generator and check matrices as follows has free distance 8.

$$G = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z + \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z^3 + \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} z^4$$

$$H = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z + \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z^3 + \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} z^4$$

6 Direct products: Turbo-effect

Examples of convolutional codes formed using α_i with $\alpha_i^2 = 0$ in FG have been produced. Consider now $F(G \times H)$ and let $w = \beta \times \alpha_i$ for *any* $\beta \in FH$. Then $w^2 = \beta^2 \alpha_i^2 = 0$. This expands enormously the range of available elements whose square is zero. Note also that over a field of characteristic 2 if $\alpha^2 = 0 = \gamma^2$ then $(\alpha + \gamma)^2 = 0$.

For example in $\mathbb{Z}_2 C_2$ the element $1 + a$ was used where C_2 generated by a. Then in $\mathbb{Z}_2(G \times C_2)$ consider $\alpha = \beta(1 + a)$ for any $\beta \in \mathbb{Z}_2 G$. Then $\alpha^2 = 0$.

A simple example of this is $\mathbb{Z}_2(C_2 \times C_2)$ where $\alpha = (1 + a)b + (1 + b)a = a + b$. The matrix of $a + b$ is $\begin{pmatrix} A & B \\ B & A \end{pmatrix}$ where $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. In forming $(4, 2)$ convolutional codes we would only use the top half of the matrices, i.e. $P = \left(\begin{array}{cc|cc} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{array}\right)$. Note that in this encoding the vector (γ, δ) is mapped to $(\gamma, \delta)P = \left(\begin{array}{cc|cc} \delta & \gamma & \gamma & \delta \end{array}\right)$. This is like an interweaving of two codes.

To get a permutation effect, use the direct product with S_n, the permutation or symmetric group on n letters.

7 (2,1) codes

See [8] for examples of $(2,1)$ optimal codes up to degree 10. These can be reproduced algebraically and properties derived using the methods developed here.

Further new $(2,1)$ convolutional codes and series of convolutional $(2,1)$ are constructed in this section as an application of the general methods described above. The free distances can often be determined algebraically and codes to a prescibed free distance can be constructed by using Theorem 7.3 below.

Let F be a field of characteristic 2 and $R = FC_2$, where C_2 is generated by a. Consider elements $\alpha_i \in R$, $i > 0$, where either $\alpha_i = 1 + a$ or $\alpha_i = 0$. Then $\alpha_i^2 = 0$.

Let $\alpha_0 = 1$ in R and define $w = \alpha_1+\alpha_0 z+\alpha_2 z^2+\ldots+\alpha_n z^n$. Then $w^2 = z^2$ and hence $w\times(w/z^2) = 1$. Thus w can be used to define a $(2,1)$ convolutional code.

More generally let t be an integer, $0 \leq t \leq n$, and define $w = \sum_{i=0}^{n} \beta_i z^i$ where $\beta_i = \alpha_i, i \neq t$, $\beta_t = 1$. Then $w^2 = z^{2t}$ gives that $w \times (w/z^{2t}) = 1$. Thus w can be used to define a convolutional $(2,1)$ code. The case $\alpha_0 = \beta_1$ is a special case.

Now determine the code by choosing the first row of the matrix of w to be the generator/encoder matrix and then the last column of w/z^{2t} is the control matrix.

The matrix of α_i is $\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ when $\alpha_i = 1 + a$ and is the zero 2×2 matrix when $\alpha_i = 0$. .

Define $\delta_i = 1$ when $\alpha_i \neq 0$ and $i \neq t$; $\delta_i = 0$ when $\alpha_i = 0$ and $i \neq t$; and define $\delta_t(1,1)$ to be $(1,0)$.

Then the encoder matrix of the code is $G = (1,1) + \delta_1(1,1)z + \delta_2(1,1)z^2 + \ldots + \delta_n(1,1)z^n$ and with $H = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \delta_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} z + \delta_2 \begin{pmatrix} 1 \\ 1 \end{pmatrix} z^2 + \ldots + \delta_n \begin{pmatrix} 1 \\ 1 \end{pmatrix} z^n$, the control matrix is H/z^{2t}.

The generator matrix G obtained in this way is noncatastrophic as it has a right finite weight inverse – see Theorems 6.3 and 6.6 in [8].

For $n = 2$ we get as an example the code with the generator matrix $G = (1,1) + (1,0)z + (1,1)z^2$. This code has free distance 5 which is optimal. It is precisely the $(2,1,2,5)$ code as described in [8], page 1085.

Theorem 7.1 *G has free distance* 5.

Proof: Consider $\sum_{i=0}^{t} \beta_i z^i G$, with $\beta_i \in \mathbb{Z}_2$ and $\beta_t \neq 0$. In determining free distance we may consider $\beta_0 \neq 0$. The coefficients of $z^0 = 1$ and z^{t+2} are $(1,1)$, and also $(1,0)$ occurs in the expression for at least one other coefficient. Thus the free distance is $2 + 2 + 1$ which is attained by G.

□

The above proof illustrates a general method for proving free distance or getting a lower bound on the free distance. For example wherever $(1,0)$ appears in a sum making up a coefficient it will contribute a distance of at least 1 as the other non-zero coefficients, all $(1,1)$, will add up to $(1,1)$ or $(0,0)$.

The check matrix for this code is $\dfrac{\begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} z + \begin{pmatrix} 1 \\ 1 \end{pmatrix} z^2}{z^2} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} z^{-1} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} z^{-2}$.

For $n \geq 3$ it may be verified directly by similar algebraic methods that the free distance is at least 6. Appropriate choices of the α_i will give bigger free distances. See Theorem 7.3 below.

For $n = 3, and \delta_2 = 1 = \delta_3$ a $(2,1,3,6)$ convolutional code is obtained which is also optimal. Thus a degree 3 optimal distance 6 is given by the encoder matrix $G = (1,1) + (1,0)z + (1,1)z^2 + (1,1)z^3$ and the control matrix is $H/z^2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} /z^2 + \begin{pmatrix} 1 \\ 0 \end{pmatrix} /z + \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} z$. It is clear that H is also a control matrix.

The next case is $(2,1,4)$ of degree 4. The optimal distance of one of these is 7. Consider $w = \alpha_1+\alpha_0 z+\alpha_1 z^3+\alpha_1 z^4$, where $\alpha_1 = 1+a$ and $\alpha_0 = 1$ in $\mathbb{Z}_2C_2$. Then $w^2 = z^2$ and thus w gives the encoder matrix and w/z^2 gives the check matrix. The encoder matrix is $G = (1,1)+(1,0)z+(1,1)z^3+(1,1)z^4$. Call this code $\mathcal{C}$.

Theorem 7.2 *The free distance of $\mathcal{C}$ is 7.*

Proof: Consider $(\sum_{i=0}^{t} \beta_i z^i)G$, with $\beta_i \in \mathbb{Z}_2$. In determining free distance we may consider $\beta_0 \neq 0$ and $\beta_t \neq 0$. The coefficients of $z^0(=1)$ and z^{t+4} are both $(1,1)$. If there are more than two non-zero β_i in the sum then $(1,0)$ occurs in at least three coefficients giving a distance of $2+2+3 = 7$ at least. It is now necessary to consider the case when there are just two β_i in the sum. It is easy to see then that at least three of the coefficients of z^i are $(1,1)$, and $(1,0)$ or $(0,1)$ is a coefficient of another. Thus the free distance is 7. $\square$

Consider the next few degrees. Let $\alpha = 1+a, \alpha_0 = 1$ in FC_2 where F has characteristic 2.

1. deg 5: $w = \alpha+\alpha_0 z+\alpha z^3+\alpha z^4+\alpha z^5$; gives a free distance of 8.
2. deg 6: $w = \alpha+\alpha z^2+\alpha z^3+\alpha_0 z^4+\alpha z^5+\alpha z^6$. This gives a free distance of 10.
3. Consider for example the following degree 12 element.
   $w = \alpha+\alpha z^2+\alpha z^4+\alpha z^5+\alpha z^6+\alpha_0 z^9+\alpha z^{10}+\alpha z^{11}+\alpha z^{12}$
   Note that this resembles the polynomial used for the Golay $(23,12)$ code – see e.g. [1] page 119. The difference is that a z^{12} has been added and the coefficient of z^9 appears with coefficient α_0 and not 0 as in the Golay code. It is possible to play around with this by placing α_0 as the coefficient of other powers of z in w.

We thus study the best performance of convolutional codes derived from $w = \sum_{i=0}^{t} \alpha_i z^i$ where some $\alpha_t = 1 \in FC_2$, and all the other α_i are either 0 or else $1+a$ in FC_2. Try to choose the α_i as one would for a linear cyclic code so as to maximise the (free) distance.

The set-up indicates we should look at existing cyclic codes and form convolutional codes by mimicking the generating polynomials for the cyclic codes.

7.1 From cyclic codes to convolutional codes

Suppose now $\mathcal{C}$ is a (linear) cyclic (n,k,d_1) code over the field F of characteristic 2. Suppose also that the dual of $\mathcal{C}$, denoted $\hat{\mathcal{C}}$, is an $(n,n-k,d_2)$ code.

Let $d = min(d_1,d_2)$. Suppose $f(g) = \sum_{i=0}^{r} \beta_i g^i$, with $\beta_i \in F, (\beta_r \neq 0)$, is a generating polynomial for $\mathcal{C}$. In $f(g)$, assume $\beta_0 \neq 0$.

Consider $f(z) = \sum_{i=1}^{r} \alpha_i z^i$ where now $\alpha_i = \beta_i\alpha$ with $\alpha = 1+a$ in FC_2 or else $\alpha_i = 0$. Replace some α_i, say α_t, by 1 or a (considered as members of FC_2).

So assume $f(z) = \sum_{i=0}^{r} \alpha_i z^i$ with this $\alpha_t = 1$ and other $\alpha_i = \beta_i\alpha$ so that $\alpha_i = 1+a$ or $\alpha_i = 0$ (for $i \neq t$). It is also allowed to let $\alpha_t = a$.

Then $f(z)^2 = z^{2t}$ and thus $f(z)\times(f(z)/z^{2t}) = 1$. We now use $f(z)$ to generate a convolutional code by taking just the first rows of the α_i. Thus the generating matrix is $\hat{f} = \sum_{i=0}^{r} \hat{\alpha}_i z^i$ where $\hat{\alpha}_i$ is the first row of α_i.

Lemma 7.1 *Let G be a generator matrix of a linear code $\mathcal{C}$ and suppose the dual code of $\mathcal{C}$, $\hat{\mathcal{C}}$, has distance d. Then no row of G is a combination of less than $d-1$ other rows of G.*

Proof: Now G^{T} is the check matrix of $\hat{\mathcal{C}}$. Since $\hat{\mathcal{C}}$ has distance d any $d-1$ columns of G^T are linearly independent – see e.g. [1], Corollary 3.2.3, page 52. Thus no column of G^{T} is a combination of less than $d-1$ other columns of G^{T}. Hence no row of G is a combination of less than $d-1$ other rows of G.

□

Lemma 7.2 *Let $w = \sum_{i=1}^{n} \alpha_i(1,1) + \alpha(1,0)$ with $\alpha \neq 0$. Then at least one component of w is not zero.*

Proof: Now $w = (\sum_{i=1}^{n} \alpha_i + \alpha, \sum_{i=1}^{n} \alpha_i)$. Since $\alpha \neq 0$ it is clear that one component of w is not zero. □

A similar result holds for $w = \sum_{i=1}^{n} \alpha_i(1,1) + \alpha(0,1)$.

For the following theorem assume the invertible element α_0 does not occur in the first or the last position of f; if it does occur in one of these positions, a similar result holds but the free distance is possibly less by 1.

Theorem 7.3 *Let $\mathcal{C}$ denote the convolutional code with generator matrix $\hat{f}$. Then the free distance of $\mathcal{C}$ is at least $d+2$.*

Proof:

Consider $w = \sum_{i=0}^{t} \beta_i z^i \hat{f}$ and we wish to show that its free distance is $\geq d+2$. In calculating the free distance of w we can assume $\beta_0 \neq 0$ and we also naturally assume $\beta_t \neq 0$. Let $fd(w)$ denote the free distance of w.

Let $w_1 = \sum_{i=0}^{t} \beta_i z^i$. The support of w_1, *supp*(w_1), is the number of non-zero β_i. Suppose then *supp*$(w_1) \geq d$. Then in w, α_0 appears with the coefficient of z^i, for at least d different i with $0 < i < t+r$. Also the coefficient of $1 = z^0$ is $\beta_0(1,1)$ and the coefficient of z^{t+r} is $\beta_t(1,1)$ and each of these have distance 2. Then by Lemma 7.2, w has free distance at least $d+2$.

Consider $f(g) = \sum_{i=0}^{r} \beta_i g^i$ and $H(g) = f(g)(\sum_{i=0}^{l} \delta_i g^i)$, with $l \leq k-1$ where k is the rank of the cyclic code. Then as this cyclic code has distance d_1, $H(g) = \sum_{i=0}^{n-1} \gamma_i g^i$ has support at least d_1. Now $H(z) = f(z)(\sum_{i=0}^{l} \delta_i z^i)$ is such that the sum of the coefficients of $z^i, z^{i+n}, \ldots$ is γ_i for each i. Hence if $\gamma_i \neq 0$, at least one of the coefficients of $z^i, z^{i+n}, \ldots$ is not 0. Since $H(g)$ has support d_1, this implies that $H(z)$ has support at least d_1. Hence w has free distance at least $(d_1-2)+2\times 2 = d_1+2 \geq d+2$ when $t \leq (k-1)$.

Assume then in w that $t \geq k$. and that *supp*$(w_1) < d$. If *supp*$(w_1) = 1$ then clearly $fd(w) \geq (r-2)+4 = r+2 \geq d+2$.

Assume by induction that a sum such as w of less than t elements with support less than d has free distance at least $d+2$.

Consider $f(g) = \sum_{i=0}^{r} \beta_i g^i$ and $H(g) = f(g)(\sum_{i=0}^{l} \delta_i g^i)$, where $t > k-1$.

Now as $\mathcal{C}$ has rank k, $f(g)g^k = \sum_{i=0}^{k-1} \delta_i f(g) g^i$. Thus multiplying through by g^{t-k} implies $f(g)g^t =$

$$\sum_{i=0}^{k-1} \delta_i g^{i+t-k} f(g) = \sum_{j=t-k}^{t-1} \delta_{j-(t-k)} f(g) g^j.$$

Now as $\hat{\mathcal{C}}$ has distance d_2 the support of $\sum_{i=1}^{k-1} \delta_i f(g) g^i$ and hence of $\sum_{j=t-k}^{t-1} \delta_{j-(t-k)} f(g) g^j$ is at least $d_2 - 1$ by Lemma 7.1.

Now $\sum_{i=0}^{t-1} \beta_i z^i \hat{f}$ has support at most $d-2$ as w has support at most $d-1$.

Then $w = \sum_{i=0}^{t} \beta_i z^i \hat{f} = \sum_{i=0}^{t-1} \beta_i z^i \hat{f} + \beta_t z^t \hat{f} = \sum_{i=0}^{t-1} \beta_i z^i \hat{f} + \beta_t \sum_{j=t-k}^{t-1} \delta_{j-(t-k)} \hat{f} z^j = \sum_{i=0}^{t-1} \omega_i \hat{f} z^i$ and this sum is of non-zero support. Thus by induction the $fd(w) \geq d+2$.

□

The free distance may be bigger than $d+2$; an upper bound is $2d-1$. The free distance also depends on where the invertible α_0 is placed in the expression for f. Placed near the 'centre' will possibly give the best free distance.

It is worth noting that if the support of the input element is $\geq t$ then the free distance is at least $t+2$; this may be seen from the proof of Theorem 7.3. Thus it is possible to avoid short distance codewords by ensuring that the input elements have sufficient support – this could be done by, for example, taking the complement of any element with small support.

The best choice for $\mathcal{C}$ is probably a self-dual code as in this case $d_1 = d_2 = d$.

There exist self-dual codes of arbitrary large distances. See also [5] for many constructions of self-dual codes.

These convolutional codes can be considered to be self-dual type convolutional codes in the sense that $f(z)$ determines the generator matrix and $f(z)/z^{2t}$ determines the control matrix.

8 (2m,1) codes

The previous section Section 7 can be generalised to produce convolutional codes of smaller rate $(2m, 1)$ but with much bigger free distance. Essentially the free distance is multiplied by m over that obtained for similar $(2, 1)$ codes.

The group to consider is C_{2m} generated by a. Assume m is odd although similar results may be obtained when m is even. Let $\alpha = 1 + a + a^2 + \ldots + a^{2m-1}$ and $\alpha_0 = 1 + a^2 + \ldots + a^{2m-2}$. Then $\alpha^2 = 0$ and $\alpha_0^2 = 1$ as α_0 has odd support.

Define as before $f(z) = \sum_{i=1}^{r} \alpha_i z^i$ where now $\alpha_i = \beta_i \alpha$ in $\mathbb{Z}_2 C_{2m}$ or else $\alpha_i = 0$. Replace some α_i, say α_t, by α_0.

Then $f(z)^2 = z^{2t}$ and $f(z)(f(z)/z^{2t}) = 1$. Thus use $f(z)$ to define a convolutional code $\mathbb{C}$ by taking the first row of the α_i.

For example $G(z) = (1,1,1,1,1,1) + (1,0,1,0,1,0)z + (1,1,1,1,1,1)z^2$ defines a $(6,1)$ convolutional code which has free distance 15. $G(z) = (1,1,1,1,1,1) + (1,0,1,0,1,0)z + (1,1,1,1,1,1)z^3 + (1,1,1,1,1,1)z^4$ defines a convolutional code which has free distance 21.

A theorem similar to Theorem 7.3 is also true:Let $f(g)$ denote the generator matrix of a cyclic code with distance d_1 and whose dual code has distance d_2. Let $d = \min(d_1, d_2)$ and let $\mathcal{C}$ denote the convolutional code obtained from $f(z)$ where the coefficients of $f(g)$ have been replaced by α_i in all but one coefficient which has been replaced by α_0 and the first row of each coefficient is used. Assume in the following theorem that α_0 is not in first or last coefficient.

Theorem 8.1 *The free distance of $\mathcal{C}$ is at least $md + 2m$.*

9 Higher rates

The methods of Section 7 can also be generalised to produce higher rate convolutional codes.

Consider achieving a rate of 3/4.

In C_4 generated by a, define $\alpha = 1 + a$ and $\alpha_0 = 1$. Then $\alpha^4 = 0$ and α can be used to define a code of rate 3/4 and distance 2. Now α has matrix

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

and the first three rows of this

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

generates a $(4, 3, 2)$ code.

Now the matrix of α_0 is $I_{4\times4}$, the identity 4×4 matrix and let B denote the first three rows of $I_{4\times4}$.

Lemma 9.1 *Let $\underline{x} \neq 0$ be a 1×3 vector. Then $\underline{x}(A + B)$ is not the zero vector and thus $\underline{x}(A + B)$ has distance at least* 1.

Proof: Now $(\alpha + 1)^4 = \alpha^4 + 1 = 1$ and so $(\alpha + 1)$ is a non-singular matrix. Thus in particular the first three rows of the matrix of $(\alpha + 1)$ are linearly independent. The first three rows of $\alpha + 1$ precisely constitutes the matrix $A + B$. Thus $\underline{x}(A + B)$ is not the zero vector.

Another way to look at this is that $\alpha + 1 = a$ but it is useful to look at the more general way in Lemma 9.1 for further developments.

□

Corollary 9.1 *If $\underline{x}A + \underline{y}B = \underline{0}$ then $\underline{x} \neq \underline{y}$.*

Form convolutional $(4, 3)$ codes as follows.

Let $f(z) = \sum_{i=0}^{n} \alpha_i z^i$ where $\alpha_i = \alpha$ or $\alpha_i = 0$ except for $\alpha_t = 1$ for some $t, 1 < t \leq n$. We could also use $\alpha_1 = \alpha_t = 1$ but this generally gives smaller distance codes.

Then $f(z)^4 = z^{4t}$ and so $f(z) \times (f(z)^3/z^{4t}) = 1$. Thus use $f(z)$ to generate the code and $(f(z)^3/z^{4t})$ to check/control the code. Take the first three rows of the matrix of $f(z)$ to generate a $(4, 3)$ code and delete the last three columns $(f(z)^3/z^{4t})$ to form the control matrix.

Thus $G(z) = \sum_{i=0}^{n} \hat{\alpha_i} z^i$ is the generator matrix where $\hat{\alpha_i}$ is the first three rows of the matrix of α_i.

In Section 7 we had the situation that when α_0 occurred in any coefficient then it contributed a distance of 1, so that when the support of G is s then α_0 will contribute a free distance of s. Here we us the fact that if α_0 occurs then it will contribute a distance of at least 1 unless its coefficient equals the sum of the coefficients in the other non-zero α_i which occur with it in the same coefficient of z^j.

9.1 Examples

The generator matrix

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} z + \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} z^2$$

defines a $(4, 3)$ convolutional code. It may be shown that its free distance is 5. The proof is similar to the proof of Theorem 7.1 but also using Lemma 9.1.

The check matrix for the code is easy to write out.

Consider $n = 3$, and

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} z + \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} z^2 + \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} z^3$$

This is a $(4, 3)$ convolutional code and its free distance is 6.

The next example is

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} z + \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} z^3 + \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} z^4$$

This has free distance 7. This may be proved similar to Theorem 7.2 using Lemma 9.1.

It is then possible to proceed as in Section 7 to investigate further degrees (memories) with rate 3/4.

9.2 Polynomial

In cases where a polynomial generator *and* polynomial right inverse for this generator are required, insist that $\alpha_0 = 1$. This gives slightly less free distance but is interesting in itself.

For example consider the encoder matrix $G = (1, 0) + \delta_1(1, 1)z + \ldots + \delta_n(1, 1)z^n$ and the control matrix is $H = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \delta_1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} z + \ldots + \delta_n \begin{pmatrix} 1 \\ 1 \end{pmatrix} z^n$. Here $\delta_i = 0$ or $\delta_i = 1$.

This code has free distance 4 for $n = 2$. For $n \geq 2$ the free distance will depend on the choice of the δ_i. As already noted, the choices where the z-polynomial corresponds to a known cyclic code polynomial deserves particular attention.

We may also increase the size of the field as for example as follows.

Consider now $R = GF(4)C_2$, the group ring of the cyclic group of order 2 over the field of 4 elements. Define $\alpha_0 = \omega + \omega^2 g$, $\alpha_1 = \omega + \omega g, \alpha_2 = \omega^2 + \omega^2 g$, where ω is the primitive element in $GF(4)$ which satisfies $\omega^2 + \omega + 1 = 0, \omega^3 = 1$. Then $\alpha_0^2 = \omega^2 + \omega^4 = \omega^2 + \omega = 1$ and $\alpha_1^2 = \alpha_2^2 = 0$. Thus $w = \alpha_0 + \alpha_1 z + \alpha_2 z^2$ satisfies $w^2 = 1$ and can be used to define a convolutional code of length 2 and dimension 1. The encoder matrix is then $G = (\omega, \omega^2) + \delta_1(\omega, \omega)z + \delta_2(\omega^2, \omega^2)z^2 + \ldots + \delta_n(\omega^i, \omega^i)z^n$ and the control matrix is $H = \begin{pmatrix} \omega^2 \\ \omega \end{pmatrix} + \delta_1 \begin{pmatrix} \omega \\ \omega \end{pmatrix} z + \ldots + \delta_n \begin{pmatrix} \omega^i \\ \omega^i \end{pmatrix} z^n$.

The *degree* of a convolutional code with encoder matrix $G(z)$ is defined to be the maximal degree of the full $k \times k$ size minors of $G(z)$ where k is the dimension; see [1]. The maximum free distance of a length 2, dimension one, degree δ code over any field is by [11], $2\delta + 2$.

Consider the case $n = 2$. The encoder matrix is then $G = (\omega, \omega^2) + (\omega, \omega)z + (\omega^2, \omega^2)z^2$. The degree of this code is $\delta = 2$ since the dimension is 1. Let $G' = (1, \omega) + (1, 1)z + (\omega, \omega)z^2$ so that $\omega G' = G$.

Theorem 9.1 *The free distance of this code is* 6 *and so is thus a maximum distance separable convolutional code.*

Proof: Consider combinations $(\alpha_0 + \alpha_1 z + \ldots + \alpha_t z^t)G$ and we wish to show that this has (free) distance 6. We may assume $\alpha_0 \neq 0$. It is clear when $t = 0$ that w has a distance of 6 and so in particular a distance of 6 is attained. Since also ω is a factor of G we may now consider the minimum distance of $w = (\alpha_0 + \alpha_1 z + \ldots + \alpha_t z^t)G'$ with $\alpha_0 \neq 0, \alpha_t \neq 0$ and $t > 0$. The coefficient of z^0 is $\alpha_0(1, \omega)$; the coefficient of z^{t+2} is $\alpha_t(\omega, \omega)$, the coefficient of z^{t+1} is $\alpha_t(1, 1) + \alpha_{t-1}(\omega, \omega)$ and the coefficient of z^t is $\alpha_t(1, \omega) + \alpha_{t-1}(1, 1) + \alpha_{t-2}(\omega, \omega)$ when $t \geq 2$ and the coefficient of z is $\alpha_1(1, \omega) + \alpha_0(1, 1)$ and this is also the case when $t = 1$.

Case $t \geq 2$: If $\alpha_t \neq \alpha_{t-1}\omega$ then the coefficient of z^{t+1} has distance 2 giving a distance of 6 with 2 coming from each of the coefficients of z^0, z^{t+1}, z^{t+2}. If $\alpha_t = \alpha_{t-1}\omega$ the coefficient of z^t is $\alpha_{t-1}(\omega + 1, \omega^2 + 1) + \alpha_{t-1}(\omega, \omega)$; in any case this has distance ≥ 1. Also the coefficient of z has distance ≥ 1. Thus the total distance is at least $2 + 1 + 1 + 2 = 6$.

Case $t=1$. If $\alpha_0\omega \neq \alpha_1$ then the coefficient of z^2 has distance 2 and thus get a distance of $2+2+2=6$ for the coefficients of z^0, z^2, z^3. If $\alpha_0\omega = \alpha_1$ then the coefficient of z is $\alpha_1(1,\omega)+\alpha_0(1,1) = \alpha_0(\omega+1, \omega^2+1)$ which has distance 2. Thus also we get a distance of $2+2+2=6$ from coefficients of z^0, z, z^3.

Note that the proof depends on the fact that $\{1,\omega\}$ is linearly independent in $GF(4)$. $\square$

9.2.1 Bigger fields

It will be necessary to work over bigger fields to get length 2, dimension 1, maximal distance separable convolutional codes of higher degree.

Consider $\mathbb{F} = GF(2^n)$ with generating element ω satisfying $\omega^n+\omega+1=0$. Then $w_0 = \omega+\omega^n a$ in $\mathbb{F}C_2$, where C_2 is generated by a satisfies $w_0^2 = \omega^2+\omega^{2n} = 1$ since $\omega^n = \omega+1$ and $w_i = \omega^i+\omega^i$, defined for $i>0$, satisfies $w_i^2 = 0$.

A generating element is then formed from these w_i. Consider $w(z) = w_0 + \delta_1 w_{i_1} z + \ldots + \delta_n w_{i_n} z^n$ where w_{i_j} is some w_i and $\delta_i \in \{0.1\}$. Then $w(z)^2 = 1$ and is then used to define a convolutional code of length 2 and dimension 1.

The w_0 can be taken as the coefficient of any z^t in the definition of $w(z)$ and convolutional codes are similarly defined.

The further study of these codes is not included here.

10 General rank considerations

Let $w(z) = \displaystyle\sum_{i=0}^{t} \alpha_i z^i$ where $\alpha_i^2 = 0, i \neq t, \alpha_t^2 = 1$ with the α_i in some group ring RG. Suppose the α_i commute and that R has characteristic 2. Then $w(z)^2 = z^{2t}$.

Consider the ranks of the non-zero α_i in deciding which rows of w to choose with which to construct the convolutional code. For example if the non-zero α_i satisfy $\operatorname{rank}\alpha_i = 1/2|G| = m$ we choose the matrix with just half the rows of the matrix of each α_i.

Many good codes may be produced this way.

It is possible to have more than one α_t satisfying $\alpha_t^2 = 1$ in $w(z)$ but then the generator matrix produced can be catastrophic, although a valid code may still be defined.

10.1 Example

Let $u = 1 + h(a+a^2+a^3)$ in $\mathbb{Z}_2(C_4 \times C_2)$. Then $u^2 = 0$ and $\operatorname{rank} u = 4$. Define $w = u + z + uz^2$. Then $w^2 = z^2$ and w is used to define a $(8,4)$ convolutional code. The generator matrix is $G = (I,B) + (I,0)z + (I,B)z^2$ where $B = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$. Now (I,B) has distance 4. Any combination of $(I,B), (I,0)$ has distance 1 at least as B is non-singular. Thus consider $(\displaystyle\sum_{i=0}^{t} \beta_i z^t)G$. The highest and lowest power of z has distance 4 and there is a power of z in between which has distance 1 so altogether we get a free distance of 9. The degree of the code is 8.

This can be extended. It can also be extended by finding higher dimensional u with $u^2 = 0$. See Section 14 for further development of these ideas.

10.2 Higher rates with nilpotent elements

So far we have used α_i with $\alpha_i^2 = 0$ and this generally give rate 1/2 convolutional codes. We now look at elements α with $\alpha^4 = 0$ with which to produce convolutional rate 3/4 codes. See [5] for where such elements are used to produce dual-containing codes.

See Section 9 for some preliminary examples on these.

Suppose then $w = \sum_{i=0}^{n} \alpha_i z^i$ in $\mathbb{F}G$ where $\alpha_i^4 = 0, i \neq t$ and $\alpha_t^4 = 1, 1 \leq t \leq n$. Suppose also $\mathbb{F}$ has characteristic 2 and that the α_i commute. Then $w^4 = z^{4t}$. Thus w is used to generate a 3/4 rate convolutional code by taking the first 3/4 of the rows of the α_i; then w^3/z^{4t} will be the control matrix using the last 1/4 of the columns of the α_i.

For examples of elements α_i with $\alpha_i^4 = 0$, see [5].

10.2.1 Example

Consider $\alpha = a + a^7 \in \mathbb{Z}_2C_8$. Then $\alpha_i^4 = 0$ and α generates an $(8, 6, 2)$ linear cyclic code − this is the best distance for a linear $(8, 6)$ code. Now construct convolutional codes similar to the construction of the $(2, 1)$ codes.

An element $\alpha_0 \in \mathbb{Z}_2C_8$ such that $\alpha_0^4 = 1$ is needed. There are a number of choices including $\alpha_0 = 1, \alpha_0 = 1 + a + a^3, \alpha_0 = 1 + a + a^7$. Choose α_0 so that the first 3 rows of the matrix of α_0 generates a linear code of largest distance. It is easy to verify that the first three rows of $\alpha_0 = 1 + a + a^3$ generates a linear code of distance 2.

- $w = \alpha + \alpha_0 z$. This gives a $(8, 6)$ code of free distance 4. The 'degree' in the convolutional sense is 6.
- $w = \alpha + \alpha_0 z + \alpha z^2$. This is a $(8, 6)$ convolutional code of free distance 6. The 'degree' here is 12.
- $w = \alpha + \alpha_0 z + \alpha z^2 + \alpha z^3$ gives an $(8, 6)$ code of free distance 6.
- $w = \alpha + \alpha_0 z + \alpha z^3 + \alpha z^4$ gives an $(8, 6)$ code of free distance 8.
- Polynomial degree 5: $w = \alpha + \alpha_0 z + \alpha z^3 + \alpha z^4 + \alpha z^5$. The free distance has to be determined.
- Polynomial degree 6: $w = \alpha + \alpha z^2 + \alpha z^3 + \alpha_0 z^4 + \alpha z^5 + \alpha z^6$. This should give a free distance of at least 10.
- As for the $(2, 1)$ convolutional codes in Section 7, by mimicking the polynomials used to generate cyclic codes, it should be possible to get $(8, 6)$ convolutional codes with increasing free distance.

11 Using idempotents to generate convolutional codes

Let FG be the group ring over a field F. For most cases in applications it is required that $\mathrm{char}\, F \not| |G|$. It may also be necessary to require that F contains a primitive n^{th} root of unity. The complex numbers $F = \mathbb{C}$ satisfies these conditions.

The reader is (again) referred to [9] for background definitions and results on group rings in relation to this section.

Let $\{e_1, e_2, \ldots, e_k\}$ be a complete family of orthogonal idempotents in FG. Such sets always exist when $\mathrm{char}\, F \not| |G|$.

Thus:
(i) $e_i \neq 0$ and $e_i^2 = e_i$, $1 \leq i \leq k$.
(ii) If $i \neq j$ then $e_i e_j = 0$.
(iii) $1 = e_1 + e_2 + \ldots + e_k$.

Here 1 is used for the identity of FG.

Theorem 11.1 *Let* $f(z) = \sum_{i=0}^{k} \pm e_i z^{t_i}$. *Then* $f(z)f(z^{-1}) = 1$.

Proof: Since $e_1, e_2, \ldots, e_k$ is a set of orthogonal primitive idempotents, $f(z)f(z^{-1}) = e_1^2 + e_2^2 + \ldots + e_k^2 = 1$. $\square$

The result in Theorem 11.1 can be considered as an identity in RC_∞ wherein $R = FG$ is a group ring.

To now construct convolutional codes, decide on the rank r and then use the first r rows of the matrices of the e_i in Theorem 11.1. The control matrix is obtained from $f(z^{-1})$ by deleting the last r columns of the e_i.

If the e_i have rank $\geq k$ and for some i $\operatorname{rank} e_i = k$ then it is probably best to take the $r = k$ for the rank of the convolutional code, although other cases also have uses depending on the application in mind.

11.1 Idempotents in group rings

Orthogonal sets of idempotents may be obtained in group rings from the conjugacy classes and character tables, see e.g. [9].

Notice also that a product $h(z) = \prod_i f_i(z)$ where the $f_i(z)$ satisfy the conditions of Theorem 11.1 also satisfies $h(z)\hat{h}(z^{-1}) = 1$, where $\hat{h}(z^{-1})$ is the product of the $f_i(z^{-1})$ in reverse order, and thus $h(z)$ can then be used to define convolutional codes.

In the ring of matrices define e_{ii} to be the matrix with 1 in the i^{th} diagonal and zeros elsewhere. Then $e_{11}, e_{22}, \ldots, e_{nn}$ is a complete set of orthogonal idempotents and can be used to define such $f(z)$. These in a sense are trivial but can be useful and can also be combined with others.

To construct convolutional codes:

- Find sets of orthogonal idempotents.
- Decide on the $f(z)$ to be used with each set.
- Take the product of the $f(z)$.
- Decide on the rate.
- Convert these idempotents into matrices as per the isomorphism between the group ring and a ring of matrices.

Group rings are a rich source of complete sets of orthogonal idempotents. This brings us into character theory in group rings. Orthogonal sets over the rationals and other fields are also obtainable.

The Computer Algebra packages GAP and Magma can construct character tables and conjugacy classes from which complete sets of orthogonal idempotents may be obtained.

11.2 Example 1

Consider $\mathbb{C}C_2$ where C_2 is generated by a. Define $e_1 = \frac{1}{2}(1+a)$ and $e_2 = 1 - e_1 = \frac{1}{2}(1-a)$. This gives $f(z) = e_1 + e_2 z^t$ or $f(z) = e_2 + e_1 z^t$ for various t. Products of these could also be used but in this case we get another of the same form by a power of z.

11.3 Cyclic

The orthogonal idempotents and character table of the cyclic group are well-known and are closely related to the Fourier matrix.

This gives for example in C_4, $e_1 = \frac{1}{4}(1+a+a^2+a^3), e_2 = \frac{1}{4}(1+\omega a+\omega^2 a^2+\omega^3 a^3), e_3 = \frac{1}{4}(1-a+a^2-a^3), e_4 = \frac{1}{4}(1+\omega^3 a+\omega^2 a^2+\omega a^3)$ from which 4×4 matrices with degree 4 in z may be constructed, where ω is a primitive 4^{th} root of unity. Notice in this case that $\omega^2 = -1$.

Let $f(z) = e_1 + e_2 z + e_3 + e_4 z^3$. Then $f(z)f(z^{-1}) = 1$. We take the first row of the matrices to give the following generator matrix for a $(4,1,3)$ convolutional code:

$G(z) = \frac{1}{4}\{(1,1,1,1) + (1,\omega,-1,-\omega)z + (1,-1,1,-1)z^2 + (1,-\omega,-1,\omega)z^3\}$.

It is easy to check that a combination of any one, two or three of the vectors $(1,1,1,1), (1,\omega,-1,-\omega), (1,-1,1,-1), (1,-\omega,-1,\omega)$, which are the rows of the Fourier matrix, has distance at least 2 and a combination of all four of them has distance 1. From this it is easy to show that

the code has free distance 14 – any combination of more than one will have 4 at each end and three in the middle with distance at least 2. This gives a $(4, 1, 3, 14)$ convolutional codes which is optimal – see [11].

We can combine the e_i to get real sets of orthogonal idempotents. Note that it is enough to combine the conjugacy classes of g and g^{-1} in order to get real sets of orthogonal idempotents.

In this case then we get

$\hat{e}_1 = e_1 = \frac{1}{4}(1 + a + a^2 + a^3), \hat{e}_2 = e_2 + e_4 = \frac{1}{2}(1 - a^2), \hat{e}_3 = e_3 = \frac{1}{4}(1 - a + a^2 - a^3)$, which can then be used to construct real convolutional codes.

Then $G(z) = \frac{1}{4}\{(1, 1, 1, 1) + 2(2, 0, -2, 0)z + (1, -1, 1, -1)z^2\}$ gives a $(4, 1, 2)$ convolutional code. Its free distance is 10 which is also optimal.

Using $C_2 \times C_2$ gives different matrices. Here the set of orthogonal idempotents consists of $e_1 = \frac{1}{4}(1 + a + b + ab), e_2 = \frac{1}{4}(1 - a + b - ab), e_3 = \frac{1}{4}(1 - a - b + ab), e_4 = \frac{1}{4}(1 + a - b - ab)$ and the matrices derived are all real.

This gives $G(z) = \frac{1}{4}\{(1, 1, 1, 1) + (1, -1, 1, -1)z + (1, -1, -1, 1)z^2 + (1, 1, -1, -1)z^3\}$. Its free distance also seems to be 14.

11.4 Symmetric group

The orthogonal idempotents of the symmetric group are well-understood and are real.

We present an example here from S_3, the symmetric group on 3 letters.

Now $S_3 = \{1, (1, 2), (1, 3), (2, 3), (1, 2, 3), (1, 3, 2)\}$ where these are cycles. We also use this listing of S_3 when constructing matrices.

There are three conjugacy classes: $K_1 = \{1\}$; $K_2 = \{(1, 2), (1, 3), (2, 3)\}$; $K_3 = \{(1, 2, 3), (1, 3, 2)\}$.

Define
$\hat{e}_1 = 1 + (1, 2) + (1, 3) + (2, 3) + (1, 2, 3) + (1, 3, 2)$,
$\hat{e}_2 = 1 - \{(1, 2) + (1, 3) + (2, 3)\} + (1, 2, 3) + (1, 3, 2)$,
$\hat{e}_3 = 2 - \{(1, 2, 3) + (1, 3, 2)\}$,

and $e_1 = \frac{1}{6}\hat{e}_1; e_2 = \frac{1}{6}\hat{e}_2; e_3 = \frac{1}{3}\hat{e}_3$. Then $\{e_1, e_2, e_3\}$ form a complete orthogonal set of idempotents and may be used to construct convolutional codes.

The G-matrix of S_3 (see [4]) is

$$\begin{pmatrix} 1 & (12) & (13) & (23) & (123) & (132) \\ (12) & 1 & (132) & (123) & (23) & (13) \\ (13) & (123) & 1 & (132) & (12) & (23) \\ (23) & (132) & (123) & 1 & (13) & (12) \\ (132) & (23) & (12) & (13) & 1 & (123) \\ (123) & (13) & (23) & (21) & (132) & 1 \end{pmatrix}.$$

Thus the matrices of e_1, e_2, e_3 are respectively

$$E_1 = \frac{1}{6}\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$E_2 = \frac{1}{6}\begin{pmatrix} 1 & -1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 \end{pmatrix}$$

$$E_3 = \frac{1}{3}\begin{pmatrix} 2 & 0 & 0 & 0 & -1 & -1 \\ 0 & 2 & -1 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & -1 & 2 & 0 & 0 \\ -1 & 0 & 0 & 0 & 2 & -1 \\ -1 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

Note that e_1, e_2 have rank 1 and that e_3 has rank 2.

12 Other characteristics

Convolutional codes over fields of arbitrary characteristic, and not just characteristic 2, may also be constructed using the general method as previously described.

The following theorem is similar to Theorem 5.2.

Theorem 12.1 *Let $R = FG$ be the group ring of a group G over a field F with characteristic p. Suppose $\alpha_i \in R$ commute and $\gamma_i \in F$. Let $w = \sum_{i=0}^{n} \alpha_i \gamma_i z^i \in RC_\infty$. Then $w^p = \gamma_t^p z^{pt}$ if and only if $\alpha_i^p = 0, i \neq t$ and $\alpha_t^p = 1$.*

The situation with $\gamma_t = 1$ is easiest to deal with and is not a great restriction.

Construct convolutional codes as follows. Find elements α_i with $\alpha_i^p = 0$ and units u with $u^p = 1$ in the group ring R. Then define elements as in Theorem 12.1 in $R[z]$ to form units in $R[z]$. Thus get $f(z)^p = \gamma_t^p z^{pt}$ and hence $f(z) \times f(z)^{p-1}/(\gamma_t^p z^{pt}) = 1$. From these units, convolutional codes are defined as described in Section 2 or Section 2.2.

Thus $f(z)$ may be used to define a convolutional code. By choosing the first r rows of the α_i considered as matrices defines a (n, r) convolutional code where $n = |G|$. The generator matrix is $\hat{f}(z) = \sum_{i=0}^{n} \hat{\alpha_i} \gamma_i z^i$ where $\hat{\alpha_i}$ denotes the first r rows of the matrix of α_i.

It is necessary to decide which rows of the matrix to choose in defining the convolutional code. This is usually decided by considering the rank(s) of the non-zero α_i.

12.1 Examples for characteristic 3

Suppose then F has characteristic 3 and consider $F(C_3 \times C_3)$ where the C_3 are generated respectively by g, h.

Define $\alpha = 1 + h(1 + g)$. Then $\alpha^3 = 0$. Define $\alpha_0 = 2 + 2h$. Then $\alpha_0^3 = 1$.

The matrix of α is $P = \begin{pmatrix} I & B & 0 \\ 0 & I & B \\ B & 0 & I \end{pmatrix}$ where I is the identity 3×3 matrix, 0 is the zero 3×3 matrix and $B = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$.

By row (block) operations P is equivalent to $\begin{pmatrix} I & 0 & -B^2 \\ 0 & I & B \\ 0 & 0 & 0 \end{pmatrix}$. Thus P has rank 6 and the matrix $Q = \begin{pmatrix} I & 0 & -B^2 \\ 0 & I & B \end{pmatrix}$ defines a block $(9, 6)$ code which indeed has distance 3.

Now define $\alpha_t = \alpha_0$ for some $0 < t < n$ and choose $\alpha_i = 0$ or $\alpha_i = \alpha$ for $i \neq t$. Define $f(z) = \sum_{i=0}^{n} \alpha_i z^i$. Then by Theorem 12.1, $f(z)^3 = z^{3t}$ and hence $f(z) \times (f(z)^2/z^{3t}) = 1$. Thus $f(z)$ may be used to define a convolutional code. Choose the first 6 rows of the α_i in $f(z)$ to define the code and thus we get a

$(9,6)$ convolutional code. The generator matrix is $\hat{f}(z) = \sum_{i=0}^{n} \hat{\alpha_i} z^i$ where $\hat{\alpha_i}$ denotes the first 6 rows of α_i, considered as a matrix.

The control matrix is obtained from $f(z)^2/z^{3t}$ using the last 3 columns of the α_i.

Lemma 12.1 *$\underline{x}\hat{\alpha_i} + \underline{y}\hat{\alpha_0}$ has distance at least 1 for 1×6 vectors $\underline{x}, \underline{y}$ with $\underline{y} \neq \underline{0}$.*

12.1.1 Specific examples for characteristic 3

Define $f(z) = \alpha + \alpha_0 z + \alpha z^2$. Then $\hat{f}(z) = \hat{\alpha} + \hat{\alpha_0} z + \hat{\alpha} z^2$ is a convolutional $(9,6)$ code of free distance 8. Define $f(z) = \alpha + \alpha_0 z + \alpha z^3 + \alpha z^4$. Then $\hat{f}(z) = \hat{\alpha} + \hat{\alpha_0} z + \hat{\alpha} z^3 + \hat{\alpha} z^4$ defines a $(9,6)$ convolutional code which has free distance 11.

A result similar to Theorem 7.3 can also be proved.

Suppose now $\mathcal{C}$ is a cyclic (n,k,d_1) code over the field F of characteristic 3. Suppose also that the dual of $\mathcal{C}$, denoted $\hat{\mathcal{C}}$, is an $(n, n-k, d_2)$ code.

Let $d = min(d_1, d_2)$. Suppose $f(g) = \sum_{i=0}^{r} \beta_i g^i$, with $\beta_i \in F, (\beta_r \neq 0)$, is a generating polynomial for $\mathbb{C}$. In $f(g)$, assume $\beta_0 \neq 0$.

Consider $f(z) = \sum_{i=1}^{r} \alpha_i z^i$ where now $\alpha_i = \beta_i \alpha$ with α as above in $F(C_3 \times C_3)$. Note that if $\beta_i = 0$ then $\alpha_i = 0$. Replace some α_i, say α_t, by α_0 (considered as members of $F(C_3 \times C_3)$).

So assume $f(z) = \sum_{i=0}^{r} \alpha_i z^i$ with this $\alpha_t = \alpha_0$ and other $\alpha_i = \beta_i \alpha$ (for $i \neq t$).

Then $f(z)^3 = \beta_t^3 z^{3t}$ giving that $f(z) \times (f(z)^2/(\beta_t^3 z^{3t}) = 1$. We now use $f(z)$ to generate a convolutional code by taking the first 6 rows of the α_i. Thus the generating matrix is $\hat{f}(z) = \sum_{i=0}^{r} \hat{\alpha_i} \beta_i z^i$ where $\hat{\alpha_i}$ consists of the first 6 rows of α for $i \neq t$ and $\hat{\alpha_t}$ consists of the first 6 rows of α_0.

For the following theorem assume the invertible element α_0 does not occur in the first or the last position of f.

Theorem 12.2 *Let $\mathcal{C}$ denote the convolutional code with generator matrix $\hat{f}$. Then the free distance of $\mathcal{C}$ is at least $d + 4$.*

13 General considerations

Suppose it is required that a degree n polynomial $f(z) = \alpha_0 + \alpha_1 z + \alpha_2 z^2 + \ldots + \alpha_n z^n$ is to have an inverse in $R[z]$. Then sufficient conditions on the α_i are obtained by formally multiplying $f(z)$ by a general $g(z)$ and making sure in the product that the coefficient of z^0 is 1 and the coefficient of z^i is 0 for $i > 0$.

If all the α_i commute (as for group rings on abelian groups), $2\alpha_i = 0$ (as in characteristic 2), and $\alpha_0^2 = 1, \alpha_i^2 = 0, \forall i \geq 1$, then $g(z) = \alpha_0 - \alpha_1 z - \alpha_2 z^2 - \ldots - \alpha_n z^n$ satisfies $f(z)g(z) = 1$. Choosing different α_i will maximise the distance.

It is easy to obtain elements α in the group ring with $\alpha^2 = 0$. Consider for example $\mathbb{Z}_2 C_{2n}$. Then $w_i = g^i + g^{n+i}$ for $0 \leq i < n$ satisfy $w_i^2 = 0$ and any combination α of the w_i satisfies $\alpha^2 = 0$. It is then a matter of choosing suitable combinations.

13.1 Nilpotent type

Many group rings R have elements α such that $\alpha^n = 0$ (and $\alpha^r \neq 0, r < n$). These can be exploited to produce convolutional codes.

13.1.1 Example

Consider $\mathbb{F}C_{14}$ where $\mathbb{F}$ has characteristic 2. Let $w_0 = 1 + g^5 + g^6 + g^{12} + g^{13}, w_1 = 1 + g^2 + g^5 + g^7 + g^9 + g^{12}, w_2 = 1 + g + g^3 + g^7 + g^8 + g^{10}$ and define $p = w_0 + w_1 z + w_2 z^2$. Then $p^2 = 1$. Since $w_i^2 = 0$ for $i \geq 1$, consider a rate of $\frac{1}{2}$. Thus consider the convolutional code with encoder matrix obtained from the first 7 rows of p and then the control matrix is obtained from the last 7 columns of p.

13.2 Further examples

Consider $\mathbb{Z}_2 C_8$ generated by g. Define $u = \alpha_0 + (1 + g^4)z + (1 + g^2)z^2 + (1 + g)z^3$ where $\alpha_0^8 = 1$. There are a number of choices for α_0, e.g. $\alpha_0 = 1 + g + g^3$.

Then $u^2 = \alpha_0^2 + (1+g^8)z + (1+g^4)z^2 + (1+g^2)z^6 = \alpha_0^2 + (1+g^4)z^2 + (1+g^2)z^6$, $u^4 = \alpha_0^4 + (1+g^4)z^{12}$ and $u^8 = 1$.

Then u can be used to define a convolutional code. Now $1 + g^4$ has rank 4 so for best results make it an $(8, 4)$ convolutional code by taking the first 4 rows of the matrices of u.

This is an $(8, 4, 9)$ convolutional code with degree/memory $\delta = 6$.

The rate could be increased but this would reduce the contribution from $(1 + g^4)$ matrix to distance essentially 0 as it has rank $= 4$. This would give a $(8, 6, 7)$ convolutional code.

To go further, consider $\mathbb{Z}_2 C_{16}$ etc. . Here use degree 6 or 3 as the largest power of z and it is then possible to get a $(16, 8, 9)$ convolutional code. As $\text{rank}(1 + g^4) = 8$ it is probably possible to construct a $(16, 12, 9)$ but details have not been worked out.

These are binary codes. Going to bigger fields should give better distances.

14 Hamming type

Set $R = \mathbb{Z}_2(C_4 \times C_2)$. Suppose C_4 is generated by a and C_2 is generated by h. Consider $\alpha_0 = 1 + h(1 + a^2)$ and $\alpha_i = 1 + h(a + a^2 + a^3)$ or $\alpha_i = 0$ for $i > 0$. Then $\alpha_0^2 = 1$ and $\alpha_i^2 = 0$. Define $w(z) = \sum_{i=0}^{n} \alpha_i z^i$ in RC_∞. By Theorem 5.1, $w^2 = 1$.

Let $A = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$, $B = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$ and I is the identity 4×4 matrix. The matrix corresponding to α_0 is then $\begin{pmatrix} I & A \\ A & I \end{pmatrix}$ and the matrix corresponding to α_i, $i \neq 0$, is either $\begin{pmatrix} I & B \\ B & I \end{pmatrix}$ or the zero matrix.

Now specify that the first 4 rows of w formulate the generator matrix of a code and then the last four columns of w formulate the control matrix. This gives a convolutional code of length 8 and dimension 4. It is easy to transform the resulting code into a systematic code.

The generator matrix is $G(z) = (I, A) + \delta_1(I, B)z + \delta_2(I, B)z^2 + \ldots + \delta_n(I, B)z^n$, where $\delta_i \in \{0, 1\}$. The control matrix is $H(z) = \begin{pmatrix} A \\ I \end{pmatrix} + \delta_1 \begin{pmatrix} B \\ I \end{pmatrix} z + \delta_2 \begin{pmatrix} B \\ I \end{pmatrix} z^2 \ldots + \delta_n \begin{pmatrix} B \\ I \end{pmatrix} z^n$.

The (I, A) may be moved to the coefficient of any z^i in which case the (natural) control matrix will need to be divided by a power of z to get the true control matrix.

This convolutional code may be considered as a Hamming type convolutional code as (I, B) is a generator matrix of the Hamming $(8, 4)$ code.

For $n = 1$ the free distance turns out to be 6; this can be proved in a similar manner to Theorem 9.1.

14.1 Example of this type

$G(z) = (I,B) + (I,A)z + (I,B)z^2$ with control matrix $H(z)/z^2$ where $H(z) = \begin{pmatrix} B \\ I \end{pmatrix} + \begin{pmatrix} A \\ I \end{pmatrix} z + \begin{pmatrix} B \\ I \end{pmatrix} z^2$ has free distance 10.

14.2 From cyclic to Hamming type

For $n \geq 2$, proceed as previously to define the polynomials by reference to corresponding cyclic linear polynomials. This will give convolutional codes of this type of increasing free distance. Note that (I,A) has distance 2, (I,B) (the Hamming Code) has distance 4, any combination of (I,A) and (I,B) has distance ≥ 1.

The following may be proved in a similar manner to Theorem 7.3.

Suppose now $\mathcal{C}$ is a cyclic (n,k,d_1) code over the field F of characteristic 2 and that the dual of $\mathcal{C}$, $\hat{\mathcal{C}}$, is an $(n,n-k,d_2)$ code. Let $d = min(d_1,d_2)$.

Assume $f(g) = \sum_{i=1}^{r} \beta_i g^i$ is a generator polynomial for $\mathcal{C}$. In $f(g)$, it is possible to arrange that $\beta_0 \neq 0$ and naturally assume that $\beta_r \neq 0$. Define $f(z) = \sum_{i=1}^{r} \alpha_i z^i$ with the $\alpha_i = \beta_i \alpha_i$, $i \neq t$ and $\alpha_t = \alpha_0$.

Then $f(z)^2 = z^{2t}$ giving $f(z) \times f(z)/z^{2t} = 1$. Now use $f(z)$ to generate a convolutional code by taking just the first four rows of the α_i. Thus the generating matrix is $G = \sum_{i=0}^{r} \hat{\alpha}_i z^i$ where $\hat{\alpha}_i$ consists of the first four rows of the matrix of α_i.

Theorem 14.1 *$\mathcal{C}$ has free distance at least $d+8$.*

References

[1] Richard E. Blahut, *Algebraic Codes for data transmission*, Cambridge University Press, 2003.

[2] Gluesing-Luerssen, Heide & Schmale, Wiland, “On Cyclic Convolutional Codes”, Acta Applicandae Mathematicae, Vol. 82, No. 2, 2004, 183-237.

[3] Paul Hurley and Ted Hurley, “Codes from zero-divisors and units in group rings”, arXiv:0710.5893.

[4] Ted Hurley, “Group rings and rings of matrices”, Inter. J. Pure & Appl. Math., 31, no.3, 2006, 319-335.

[5] Ted Hurley, “Self-dual, dual-containing and related quantum codes from group rings”, preprint available at http:

[6] Paul Hurley and Ted Hurley, “ Module codes in group rings”, ISIT2007, Nice, 1981-1985, 2007.

[7] van Lint, J.H. and Wilson, R.M., *A course in Combinatorics*, Cambridge University Press, 2001.

[8] R. J. McEliece, “The algebraic theory of convolutional codes”, in *Handbook of Coding Theory, Volume I*, North Holland, Elsevier Science, 1998.

[9] César Milies & Sudarshan Sehgal, *An introduction to Group Rings*, Klumer, 2002.

[10] David J. C. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 2003.

[11] J. Rosenthal & R. Smarandache, “ Maximum distance separable convolutional codes”, Appl. Algebra Engrg. Comm. Comput. 10 (1), 15-37, 1999.

[12] R. Smarandache, H. Gluesing-Luerssen, J. Rosenthal, “Constructions for MDS-convolutional codes”, IEEE Trans. Inform. Theory, vol. IT-47, 2045-2049, 2001.

[13] F.J.MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North Holland, 1977.

National University of Ireland, Galway
Galway
Ireland.